AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the application:

LISTING OF CLAIMS:

1. (currently amended): A method for communication between a first computer and a second computer, each of which is connected to a server computer, the method comprising:

under control of a first client application at the first computer,

creating a message, wherein the message comprises at least one out of a group of: an event notification with zero text and zero content identifiers, a text message, and a content identifier; and

putting the message into a message queue; and

under control of a second client application at the second computer, retrieving the message from the message queue,

wherein the first computer, the second computer, and the server computer form part of a federated content management system and wherein the federated content management system further comprises heterogeneous servers connected to the server computer.

2. (original): The method of claim 1, wherein text comprises a string of alphanumeric characters.

3. (currently amended): The method of claim 1, wherein a the content identifier comprises an item identifier and a server name.

4. (original): The method of claim 1, wherein the message comprises an event

notification with zero text and zero content identifiers.

5. (original): The method of claim 1, wherein the message comprises text with zero

content identifiers.

6. (currently amended): The method of claim 1, wherein the message comprises zero text

and one or more content identifiers that represent items in a data store connected to at least one

of the heterogeneous serversserver computer.

7. (original): The method of claim 1, wherein the message comprises an object.

8. (original): The method of claim 1, wherein the message is put into the message queue

via a method of a class.

9. (original): The method of claim 1, wherein the message is retrieved from the message

queue via a method of a class.

10. (currently amended): An apparatus for communication between computers,

comprising:

a first computer connected to a server computer;

a second computer connected to the first computer and to the server computer in a

datastore management system; and

one or more computer programs, performed by the first and second computers, for:

under control of a first client application at the first computer,

creating a message, wherein the message comprises at least one out of a

group of: an event notification with zero text and zero content identifiers, text, and content

identifier; and

putting the message into a message queue; and

under control of a second client application at the second computer, retrieving the

message from the message queue,

wherein the first computer, the second computer, and the server computer form part of a

federated content management system and wherein the federated content management system

further comprises heterogeneous servers connected to the server computer.

11. (original): The apparatus of claim 10, wherein text comprises a string of

alphanumeric characters.

12. (original): The apparatus of claim 10, wherein a content identifier comprises an item

identifier and a server name.

13. (original): The apparatus of claim 10, wherein the message comprises an event

notification with zero text and zero content identifiers.

14. (original): The apparatus of claim 10, wherein the message comprises text with zero content identifiers.

15. (original): The apparatus of claim 10, wherein the message comprises zero text and one or more content identifiers that represent items in a data store connected to the server computer.

16. (original): The apparatus of claim 10, wherein the message comprises an object.

17. (original): The apparatus of claim 10, wherein the message is put into the message queue via a method of a class.

18. (original): The apparatus of claim 10, wherein the message is retrieved from the message queue via a method of a class.

19. (currently amended): An article of manufacture comprising a program storage medium readable by a computer and embodying one or more instructions executable by the computer to perform method steps for communication between a first computer and a second computer, each of which is connected to a server computer, comprising:

under control of a first client application at the first computer,

creating a message, wherein the message comprises at least one out of the group of event notification with zero text and zero content identifiers, text, and content identifier; and

putting the message into a message queue; and

under control of a second client application at the second computer, retrieving the message from the message queue,

wherein said first and second computers and said server are in a datastore management system,

wherein the first computer, the second computer, and the server computer form part of a federated content management system and wherein the federated content management system further comprises heterogeneous servers connected to the server computer.

20. (original): The article of manufacture of claim 19, wherein text comprises a string of alphanumeric characters.

21. (original): The article of manufacture of claim 19, wherein a content identifier comprises an item identifier and a server name.

22. (original): The article of manufacture of claim 19, wherein the message comprises an event notification with zero text and zero content identifiers.

23. (original): The article of manufacture of claim 19, wherein the message comprises text with zero content identifiers.

24. (original): The article of manufacture of claim 19, wherein the message comprises zero text and one or more content identifiers that represent items in a data store connected to the server computer.

25. (original): The article of manufacture of claim 19, wherein the message comprises an object.

26. (original): The article of manufacture of claim 19, wherein the message is put into the message queue via a method of a class.

27. (original): The article of manufacture of claim 19, wherein the message is retrieved from the message queue via a method of a class.

28. (currently amended): A system for communication between a first computer and a second computer, both connected to at least one server computer, the ~~method~~ system comprising:

under control of a first application at the first computer:

creating a message, wherein the message comprises at least one out of: an event notification, text and a content identifier, and

putting the message into a message queue; and

7

under control of a second application at the second computer,

retrieving the message from the message queue,

wherein when a body of said message comprises said text, said text is passed to the

second application, and when the body of said message comprises said content identifier, at least

one object identified by said content identifier is forwarded to the second application, and when

the body of said message comprises no said text and no said content identifier, the message is an

comprises only said event notification notifying the second application of an occurrence of an

event.

29. (previously presented): The system according to claim 28, wherein said content

identifier identifies a search result of a search performed by said first application, and wherein

said search result comprises at least one object stored in said at least one server computer.

30. (previously presented): The system according to claim 29, wherein the system is a

federated content management system.

31. (previously presented): The system according to claim 28, wherein said first and

second applications are client applications.

32. (previously presented): The system according to claim 28, wherein the system is a

distributed computing system and wherein said server connects to at least one data storage.

33. (previously presented): The system according to claim 28, wherein said first and said

second computers execute portals for messaging between said first and second applications.

34. (previously presented): The system according to claim 28, wherein said content

identifier in the body of said message is a unique item identifier and a server name.

35. (previously presented): A method for communication between a first computer and a

second computer, each of which is connected to a server computer, the method comprising:

under control of a first application at the first computer,

creating a message, wherein the message comprises a text length value and a

content identifier count value; and

putting the message into a message queue; and

under control of a second application at the second computer, retrieving the message

from the message queue,

wherein said text length value identifies length of text included in said message, and

wherein the content identifier count value identifies a number of content identifiers in said

message.

36. (canceled).

37. (currently amended): The method according to claim 35 ~~36~~, wherein when the text

length value is zero and when the content identifier count value is zero, the message is an event

notification.

38. (currently amended): The method according to claim 35 ~~36~~, wherein when the

content identifier count value is greater than zero, the message further comprises at least one

content identifier identifying an object from a heterogeneous storage.

39. (previously presented): The method according to claim 1, wherein under said control

of the first client application, the first computer connects to a queue manager located on the

server computer and puts the message into the message queue, and wherein under said control of

the second client application, the second computer connects to the same queue manager located

on the server computer and retrieves the message from the message queue.

40. (canceled).

41. (new): The method according to claim 1, wherein the heterogeneous servers are

connected to heterogeneous datastores forming part of the federated content management system

and wherein the heterogeneous datastores store multimedia data and are combined into a

consistent and unified conceptual view.

42. (new): The method according to claim 41, wherein the message comprises search results from a search executed by the first computer searching for a specified contents in the multimedia data of the heterogeneous datastores.

43. (new): The method according to claim 42, wherein the first application is a browser on the first computer.

44. (new): The method of claim 3, wherein the server name identifies one of the heterogeneous servers that is connected to a data store storing an item corresponding to the item identifier.

45. (new): The system according to claim 28, wherein said creating of the message comprises generating the body of the message with said content identifier when the message is the content identifier and wherein said creating of the message comprises generating an empty body of the message when the message is said event notification.